

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1-57. (Cancelled)

58. (currently amended) A method of facilitating redirection of traffic between a server and a client to between the client and a selected one from a plurality of replicas, the method comprising:

receiving a packet from a client, the packet having a destination identifier associated with a server;

when the packet is a start packet, at the client side, adding a tag to the start packet to indicate that the start packet should be forwarded by a device other than a client side device to ~~any~~ a plurality of replicas that each duplicates the data content of the server, wherein the tag is an option byte having one of two states that indicate whether redirection is permissible or impermissible;

storing the destination identifier of the start packet;

after storing the destination identifier of the start packet and tagging the start packet, sending the start packet to the server;

when a first acknowledgement packet associated with the start packet is received first with respect to any other acknowledgement packets, storing and associating a source identifier of the first acknowledgement packet with the stored destination identifier of the start packet;

after storing and associating the source identifier of the first acknowledgement packet, sending the first acknowledgement packet to the client;

prior to storing and associating the source identifier of the first acknowledgement packet, cracking the first acknowledgement packet to obtain the source identifier when the first acknowledgement packet has been encapsulated;

when cracked, encapsulating the cracked acknowledgement packet with the destination identifier stored for the associated start packet, wherein the encapsulated first acknowledgement packet is sent to the client; and

when a second acknowledgement packet associated with the start packet is received after the first acknowledgement packet, inhibiting sending of the second acknowledgement to the client.

59. (previously presented) A method as recited in claim 58, further comprising:

when a subsequent packet associated with the start packet is received that is not a start packet or an acknowledgement packet, replacing a destination identifier of the subsequent packet with the source identifier stored for the acknowledgement packet when the subsequent packet originates from the client; and

forwarding the altered subsequent packet to its destination.

60. (previously presented) A method as recited in claim 59, wherein the subsequent packet is only modified when the destination identifier of the subsequent packet equals the destination identifier stored for the associated start packet.

61. (previously presented) A method as recited in claim 58, wherein the start packet is only tagged when the start packet is associated with web data, and the method further comprising sending the start packet to the server without the tag when the start packet is not associated with web data.

62. (previously presented) A method as recited in claim 61, wherein the start packet is associated with web data when the start packet has a destination port utilized for accessing web data.

63. (currently amended) A computer system operable to facilitate redirection of traffic between a server and a client to between the client and a selected one from a plurality of replicas, the computer system comprising:

a memory; and

a processor coupled to the memory,

wherein at least one of the memory and the processor are adapted to provide:

receiving a packet from a client, the packet having a destination identifier associated with a server;

when the packet is a start packet, at the client side, adding a tag to the start packet to indicate that the start packet should be forwarded by a device other than a client side device to any a plurality of replicas that each duplicates the data content of the server, wherein the tag is an option byte having one of two states that indicate whether redirection is permissible or impermissible;

storing the destination identifier of the start packet;

after storing the destination identifier of the start packet and tagging the start packet, sending the start packet to the server;

when a first acknowledgement packet associated with the start packet is received first with respect to any other acknowledgement packets, storing and associating a source identifier of the first acknowledgement packet with the stored destination identifier of the start packet;

after storing and associating the source identifier of the first acknowledgement packet, sending the first acknowledgement packet to the client;

prior to storing and associating the source identifier of the first acknowledgement packet, cracking the first acknowledgement packet to obtain the source identifier when the first acknowledgement packet has been encapsulated;

when cracked, encapsulating the cracked acknowledgement packet with the destination identifier stored for the associated start packet, wherein the encapsulated first acknowledgement packet is sent to the client; and

when a second acknowledgement packet associated with the start packet is received after the first acknowledgement packet, inhibiting sending of the second acknowledgement to the client.

64. (previously presented) A computer system as recited in claim 63, wherein at least one of the memory and the processor are further adapted to provide:

when a subsequent packet associated with the start packet is received that is not a start packet or an acknowledgement packet, replacing a destination identifier of the subsequent packet with the source identifier stored for the acknowledgement packet when the subsequent packet originates from the client; and

forwarding the altered subsequent packet to its destination.

65. (previously presented) A computer system as recited in claim 63, wherein the subsequent packet is only modified when the destination identifier of the subsequent packet equals the destination identifier stored for the associated start packet.

66. (previously presented) A computer system as recited in claim 63, wherein the start packet is only tagged when the start packet is associated with web data, and the method further

comprising sending the start packet to the server without the tag when the start packet is not associated with web data.

67. (previously presented) A computer system as recited in claim 66, wherein the start packet is associated with web data when the start packet has a destination port utilized for accessing web data.

68. (currently amended) A computer program product for facilitating redirection of traffic between a server and a client to between the client and a selected one from a plurality of replicas, the computer program product comprising:

at least one computer readable medium;

computer program instructions stored within the at least one computer readable product configured for:

receiving a packet from a client, the packet having a destination identifier associated with a server;

when the packet is a start packet, at the client side, adding a tag to the start packet to indicate that the start packet should be forwarded by a device other than a client side device to any a plurality of replicas that each duplicates the data content of the server, wherein the tag is an option byte having one of two states that indicate whether redirection is permissible or impermissible;

storing the destination identifier of the start packet;

after storing the destination identifier of the start packet and tagging the start packet, sending the start packet to the server;

when a first acknowledgement packet associated with the start packet is received first with respect to any other acknowledgement packets, storing and

associating a source identifier of the first acknowledgement packet with the stored destination identifier of the start packet;

after storing and associating the source identifier of the first acknowledgement packet, sending the first acknowledgement packet to the client;

prior to storing and associating the source identifier of the first acknowledgement packet, cracking the first acknowledgement packet to obtain the source identifier when the first acknowledgement packet has been encapsulated;

when cracked, encapsulating the cracked acknowledgement packet with the destination identifier stored for the associated start packet, wherein the encapsulated first acknowledgement packet is sent to the client; and

when a second acknowledgement packet associated with the start packet is received after the first acknowledgement packet, inhibiting sending of the second acknowledgement to the client.

69. (previously presented) A computer program product as recited in claim 68, the at least one computer readable product further configured for:

when a subsequent packet associated with the start packet is received that is not a start packet or an acknowledgement packet, replacing a destination identifier of the subsequent packet with the source identifier stored for the acknowledgement packet when the subsequent packet originates from the client; and

forwarding the altered subsequent packet to its destination.

70. (previously presented) A computer program product as recited in claim 69, wherein the subsequent packet is only modified when the destination identifier of the packet equals the destination identifier of the start packet.

71. (previously presented) A computer program product as recited in claim 68, wherein the start packet is only tagged when the start packet is associated with web data, and the method further comprising sending the start packet to the server without the tag when the start packet is not associated with web data.

72. (previously presented) A computer program product as recited in claim 71, wherein the start packet is associated with web data when the start packet has a destination port utilized for accessing web data.

73. (currently amended) A method of facilitating redirection of traffic between a server and a client to between the client and a nearest replica selected from a plurality of replicas, the method comprising:

at the client side, receiving a packet that is traveling between a client and a server or between the client and a replica;

when the received packet is a start packet that is traveling from the client to the server, at the client side, altering adding a tag to the start packet to indicate that the start packet should be forwarded by a device other than a client side device to ~~any~~ a plurality of replicas that each duplicates the data content of the server, wherein the tag is an option byte having one of two states that indicate whether redirection is permissible or impermissible;

when the received packet is an acknowledgement packet that is received first and spoofs the server, obtaining a source identifier of the replica from the acknowledgement when the acknowledgement originates from the replica and then sending the acknowledgement packet to the client;

when the received packet is an acknowledgement packet that is not received first and spoofs the server, inhibiting sending of the second acknowledgement to the client; and

when the received packet is a subsequent packet received after the start packet and the acknowledgement packet, altering the subsequent packet so that it goes to the replica when the subsequent packet originates from the client, wherein the alteration is based on the obtained source identifier from the acknowledgement packet.

74. (previously presented) A method as recited in claim 73, wherein the source identifier of the replica is obtained from the acknowledgement packet by cracking the acknowledgement packet when it is encapsulated.

75. (previously presented) A method as recited in claim 74, further comprising re-encapsulating the cracked acknowledgement packet prior to sending it to the client.

76. (cancelled)

77. (previously presented) A method as recited in claim 73, wherein the subsequent packet is altered by replacing the subsequent packet's destination identifier with a destination identifier of the start packet.

78. (previously presented) A method as recited in claim 73, wherein the subsequent packet is altered by encapsulating the subsequent packet with a destination identifier of the start packet.

79. (currently amended) A computer system operable to facilitate redirection of traffic between a server and a client to between the client and a nearest replica selected from a plurality of replicas, the computer system comprising:



a memory; and  
a processor coupled to the memory,  
wherein at least one of the memory and the processor are adapted to provide:

at the client side, receiving a packet that is traveling between a client and a server or between the client and a replica;

when the received packet is a start packet that is traveling from the client to the server, at the client side, ~~altering~~ adding a tag to the start packet to indicate that the start packet should be forwarded by a device other than a client side device to ~~any~~ a plurality of replicas that each duplicates the data content of the server, wherein the tag is an option byte having one of two states that indicate whether redirection is permissible or impermissible;

when the received packet is an acknowledgement packet that is received first and spoofs the server, obtaining a source identifier of the replica from the acknowledgement when the acknowledgement originates from the replica and then sending the acknowledgement packet to the client;

when the received packet is an acknowledgement packet that is not received first and spoofs the server, inhibiting sending of the second acknowledgement to the client; and

when the received packet is a subsequent packet received after the start packet and the acknowledgement packet, altering the subsequent packet so that it goes to the replica when the subsequent packet originates from the client, wherein the alteration is based on the obtained source identifier from the acknowledgement packet.

80. (previously presented) A computer system as recited in claim 79, wherein the source identifier of the replica is obtained from the acknowledgement packet by cracking the acknowledgement packet when it is encapsulated.

81. (previously presented) A computer system as recited in claim 80, further comprising re-encapsulating the cracked acknowledgement packet prior to sending it to the client.

82. (cancelled)

83. (previously presented) A computer system as recited in claim 79, wherein the subsequent packet is altered by replacing the subsequent packet's destination identifier with a destination identifier of the start packet.

84. (previously presented) A computer system as recited in claim 79, wherein the subsequent packet is altered by encapsulating the subsequent packet with a destination identifier of the start packet.

85. (currently amended) A computer program product for facilitating redirection of traffic between a server and a client to between a server and a client to between the client and a nearest replica selected from a plurality of replicas, the computer program product comprising:

at least one computer readable medium;

computer program instructions stored within the at least one computer readable product configured for:

at the client side, receiving a packet that is traveling between a client and a server or between the client and a replica;

when the received packet is a start packet that is traveling from the client to the server, at the client side, ~~altering~~ adding a tag to the start packet to indicate

that the start packet should be forwarded by a device other than a client side device to ~~any~~ a plurality of replicas that each duplicates the data content of the server, wherein the tag is an option byte having one of two states that indicate whether redirection is permissible or impermissible;

when the received packet is an acknowledgement packet that is received first and spoofs the server, obtaining a source identifier of the replica from the acknowledgement when the acknowledgement originates from the replica and then sending the acknowledgement packet to the client;

when the received packet is an acknowledgement packet that is not received first and spoofs the server, inhibiting sending of the second acknowledgement to the client; and

when the received packet is a subsequent packet received after the start packet and the acknowledgement packet, altering the subsequent packet so that it goes to the replica when the subsequent packet originates from the client, wherein the alteration is based on the obtained source identifier from the acknowledgement packet.

86. (previously presented) A computer program product as recited in claim 85, wherein the source identifier of the replica is obtained from the acknowledgement packet by cracking the acknowledgement packet when it is encapsulated.

87. (previously presented) A computer program product as recited in claim 86, the at least one computer readable product being further configured for re-encapsulating the cracked acknowledgement packet prior to sending it to the client.

88. (cancelled)

89. (previously presented) A computer program product as recited in claim 85, wherein the subsequent packet is altered by replacing the subsequent packet's destination identifier with a destination identifier of the start packet.

90. (previously presented) A computer program product as recited in claim 85, wherein the subsequent packet is altered by encapsulating the subsequent packet with a destination identifier of the start packet.

91. (currently amended) An apparatus product for facilitating redirection of traffic between a server and a client to between the client and a selected one from a plurality of replicas, the apparatus comprising:

means for receiving a packet from a client, the packet having a destination identifier associated with a server;

means for when the packet is a start packet, at the client side, adding a tag to the start packet to indicate that the start packet should be forwarded by a device other than a client side device to ~~any~~ a plurality of replicas that each duplicates the data content of the server, wherein the tag is an option byte having one of two states that indicate whether redirection is permissible or impermissible;

means for storing the destination identifier of the start packet;

means for after storing the destination identifier of the start packet and tagging the start packet, sending the start packet to the server;

means for when a first acknowledgement packet associated with the start packet is received first with respect to any other acknowledgement packets, storing and associating a source identifier of the first acknowledgement packet with the stored destination identifier of the start packet;

means for after storing and associating the source identifier of the first acknowledgement packet, sending the first acknowledgement packet to the client;

means for prior to storing and associating the source identifier of the first acknowledgement packet, cracking the first acknowledgement packet to obtain the source identifier when the first acknowledgement packet has been encapsulated;

means for when cracked, encapsulating the cracked acknowledgement packet with a source address associated with the packet, wherein the encapsulated first acknowledgement packet is sent to the client; and

means for when a second acknowledgement packet associated with the start packet is received after the first acknowledgement packet, inhibiting sending of the second acknowledgement to the client.

92. (currently amended) An apparatus product for facilitating redirection of traffic between a server and a client to between the client and a nearest replica selected from a plurality of replicas, the apparatus comprising:

means for at the client side, receiving a packet that is traveling between a client and a server or between the client and a replica;

means for when the received packet is a start packet that is traveling from the client to the server, at the client side, ~~altering~~ adding a tag to the start packet to indicate that the start packet should be forwarded by a device other than a client side device to ~~any~~ a plurality of replicas that each duplicates the data content of the server, wherein the tag is an option byte having one of two states that indicate whether redirection is permissible or impermissible;

means for when the received packet is an acknowledgement packet that is received first and spoofs the server, obtaining a source identifier of the replica

from the acknowledgement when the acknowledgement originates from the replica and then sending the acknowledgement packet to the client;

means for when the received packet is an acknowledgement packet that is not received first and spoofs the server, inhibiting sending of the second acknowledgement to the client; and

means for when the received packet is a subsequent packet received after the start packet and the acknowledgement packet, altering the subsequent packet so that it goes to the replica when the subsequent packet originates from the client, wherein the alteration is based on the obtained source identifier from the acknowledgement packet.